AI Assisted Coding

**ASSIGNMENT-11.4**

Name: Gopi chand

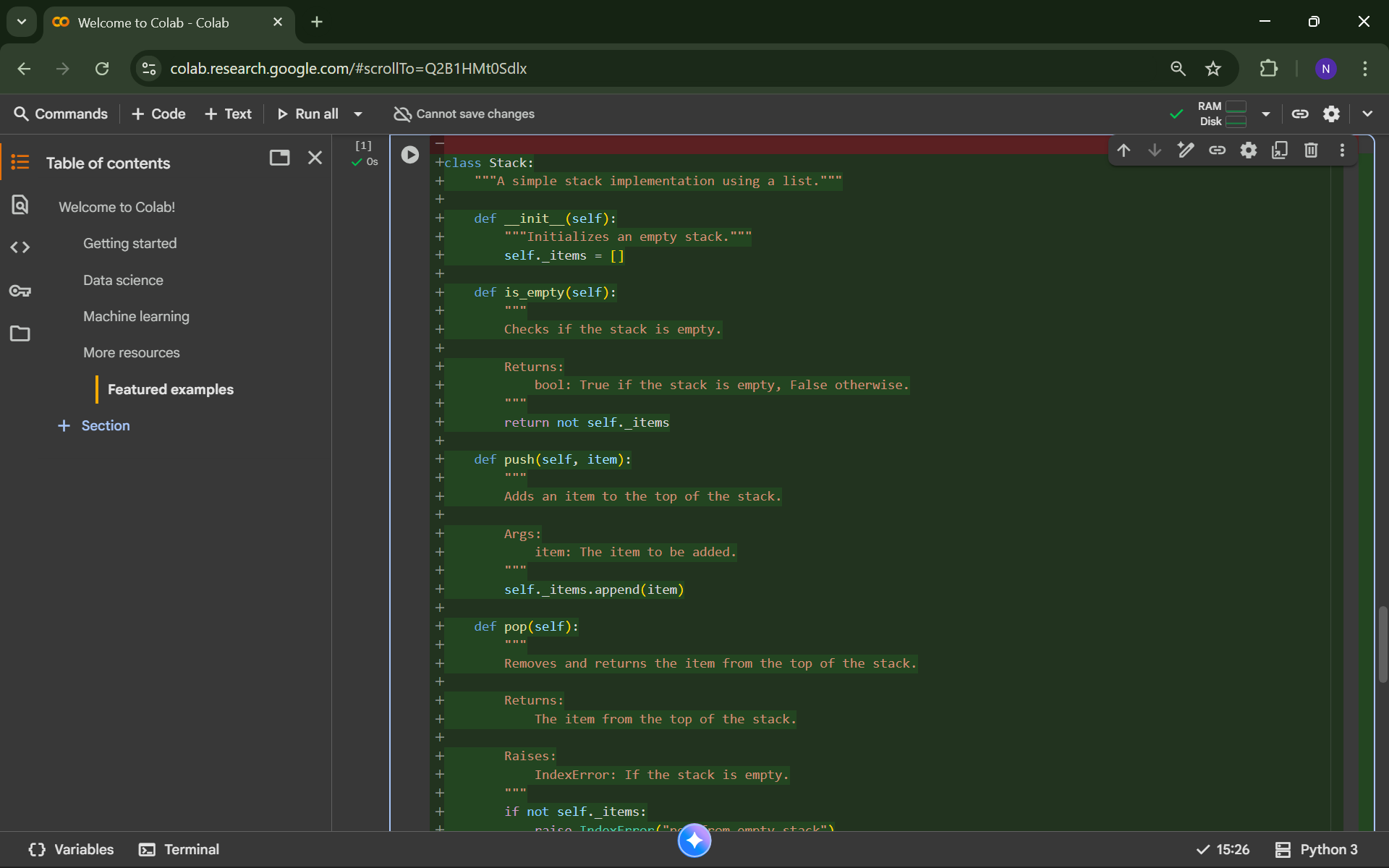
Roll no: 2403a51237

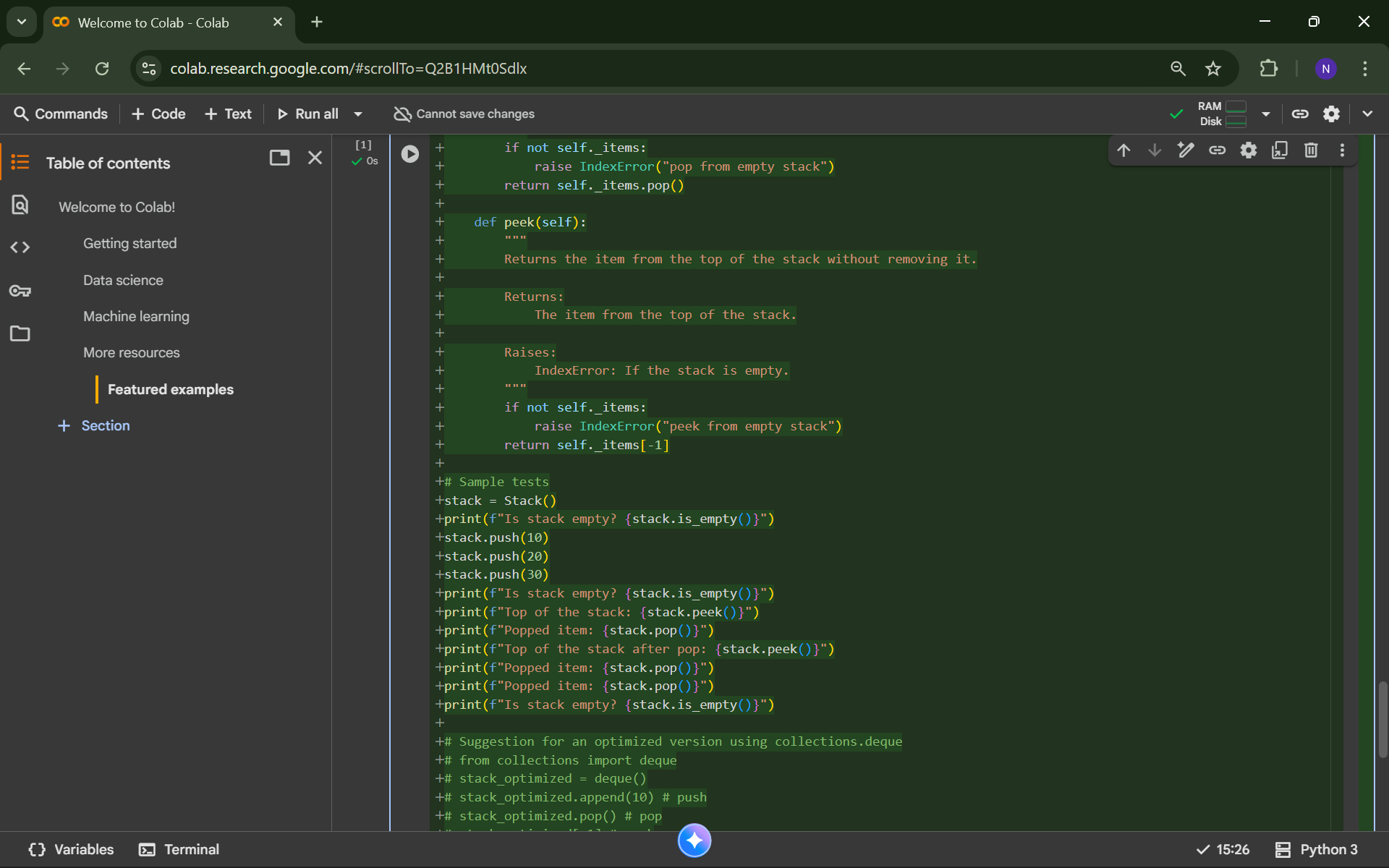
Batch no: 11

**Task 1: Implementing a Stack (LIFO)**

**PROMPT:**

1. Write a Python Stack class with push(), pop(), peek(), and is\_empty(), using Google-style docstrings and inline comments. Show sample tests with data and suggest an optimized version using collections.deque

**CODE:**



**OUTPUT:**

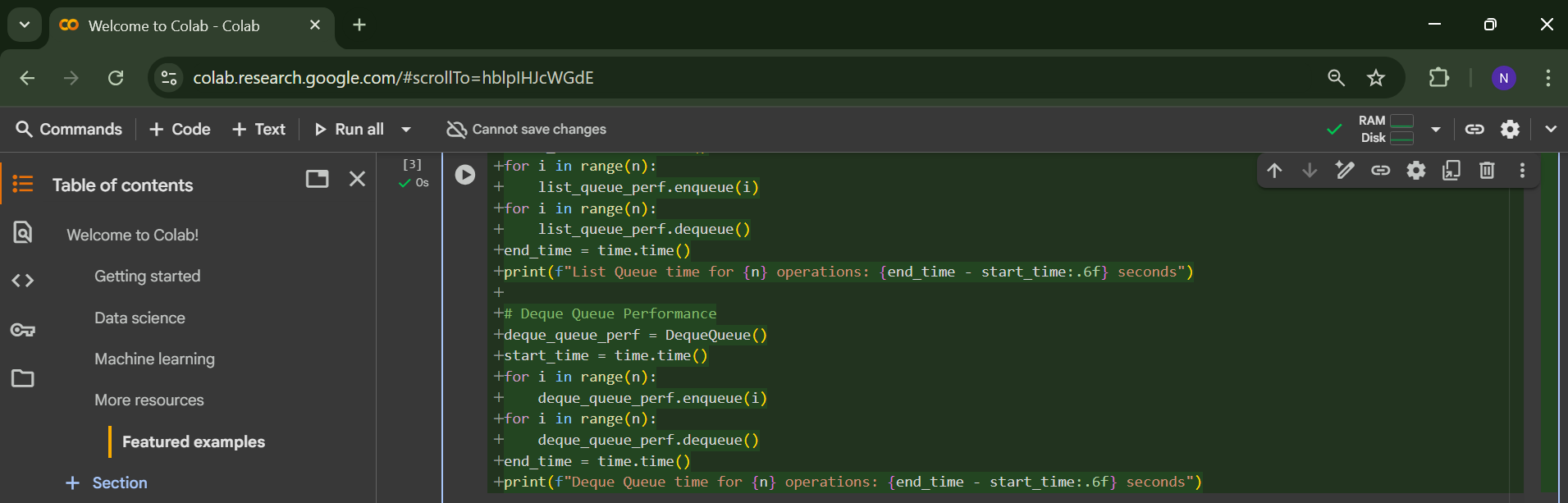
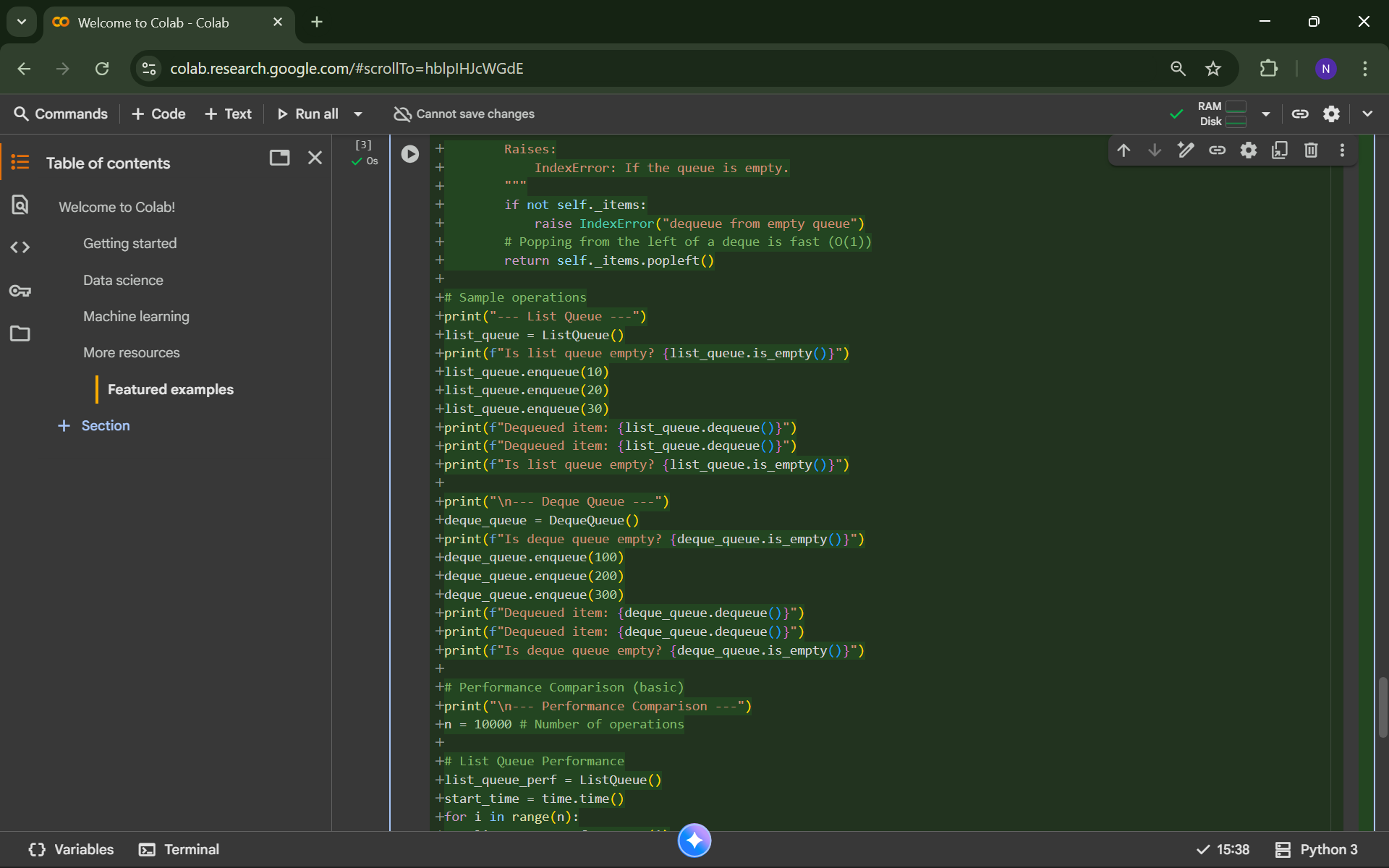
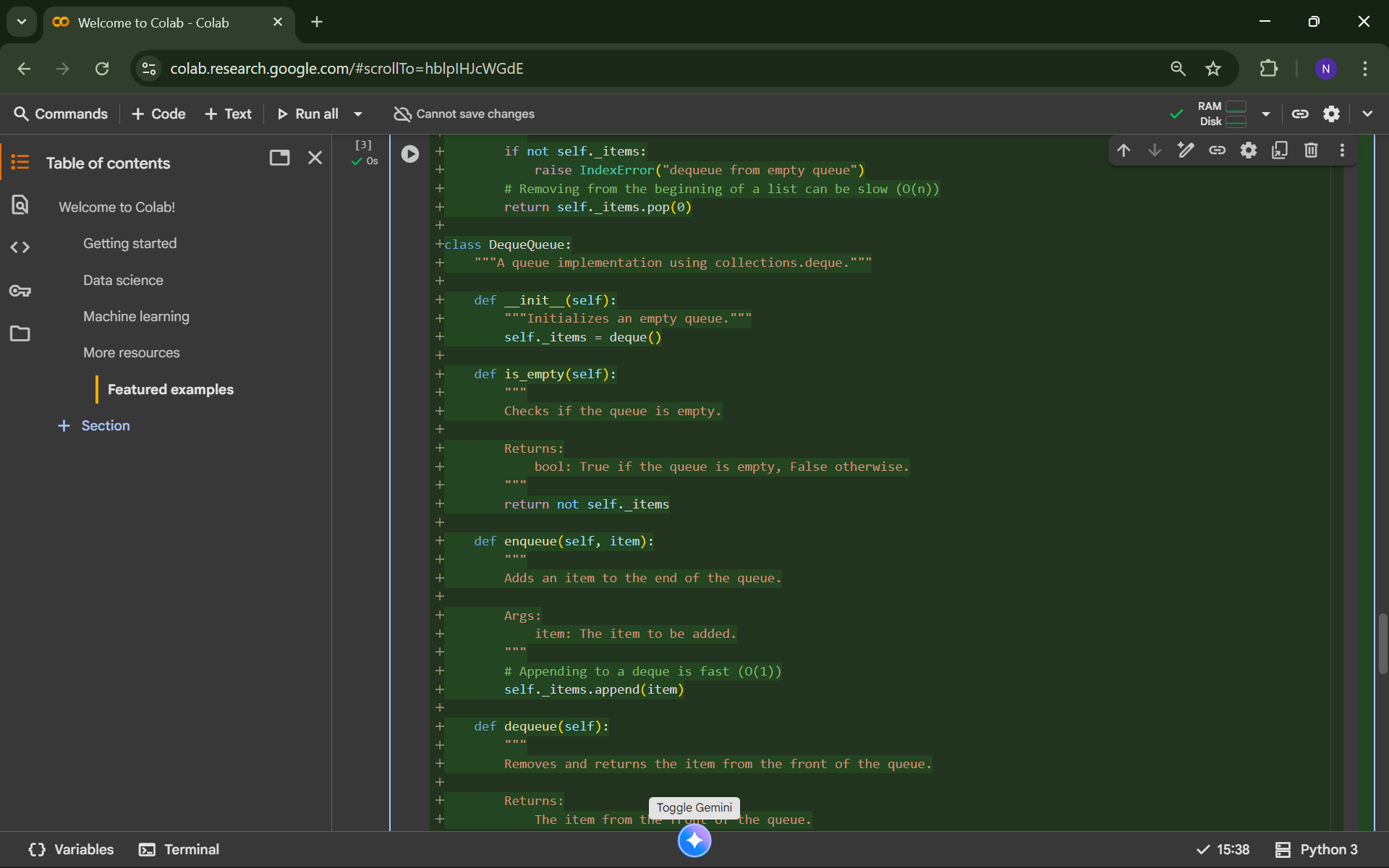
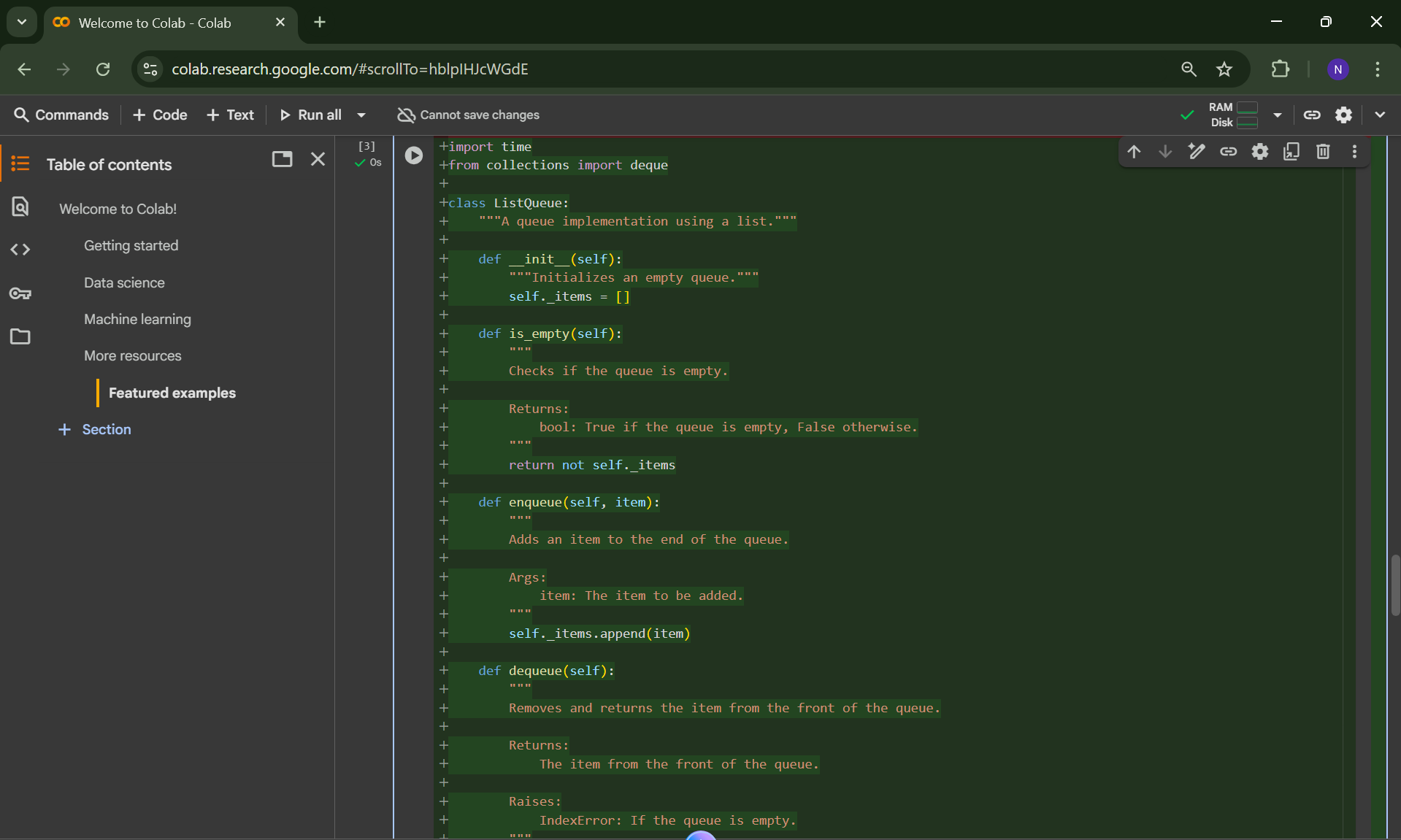


**Task 2: Queue Implementation with Performance Review**

**PROMPT:**

1. Implement two versions of a Python Queue with enqueue(), dequeue(), and is\_empty(): one with lists and one with collections.deque. Include docstrings, comments, and compare performance. Show sample enqueue and dequeue operations

**CODE:**



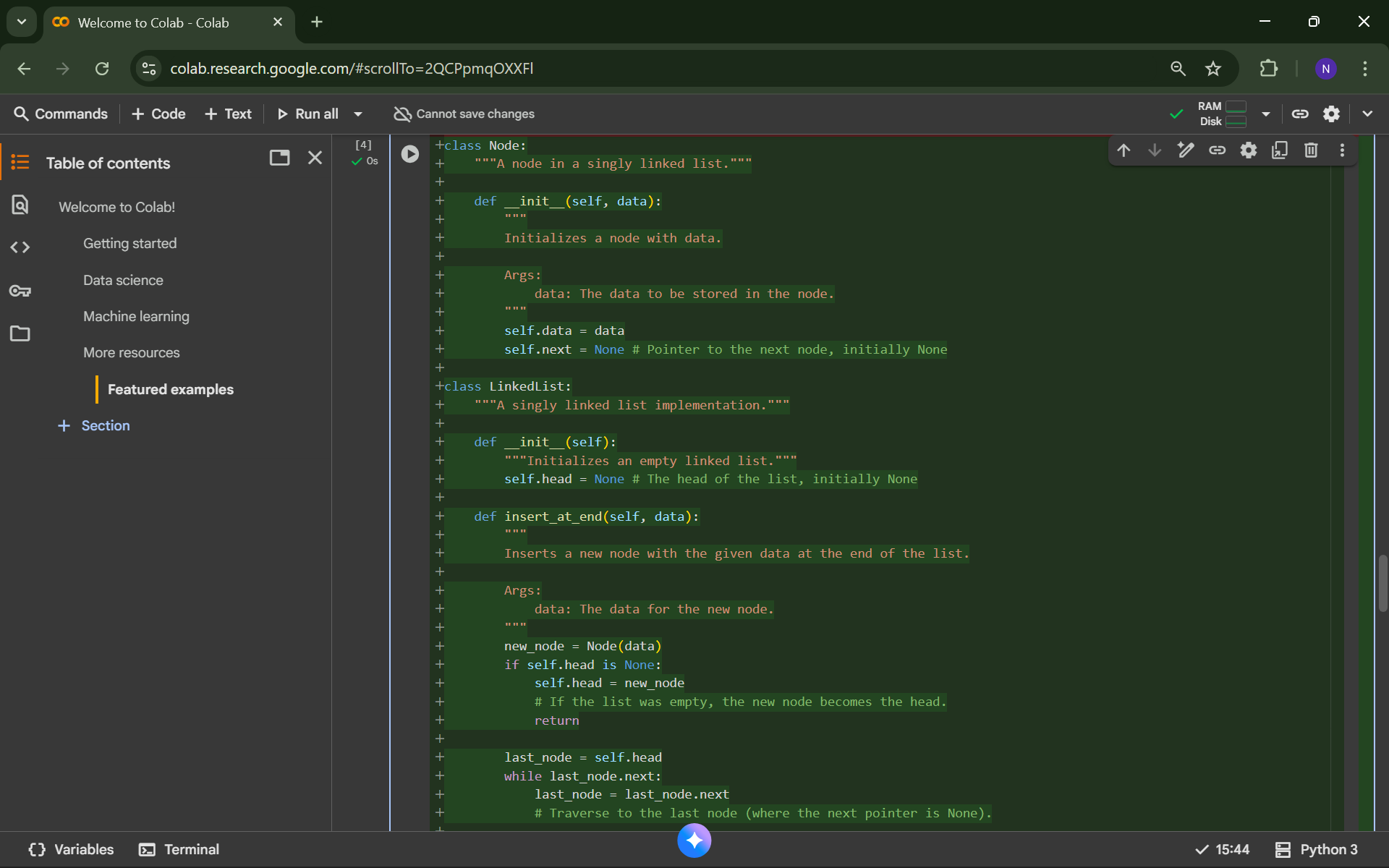
**OUTPUT:**

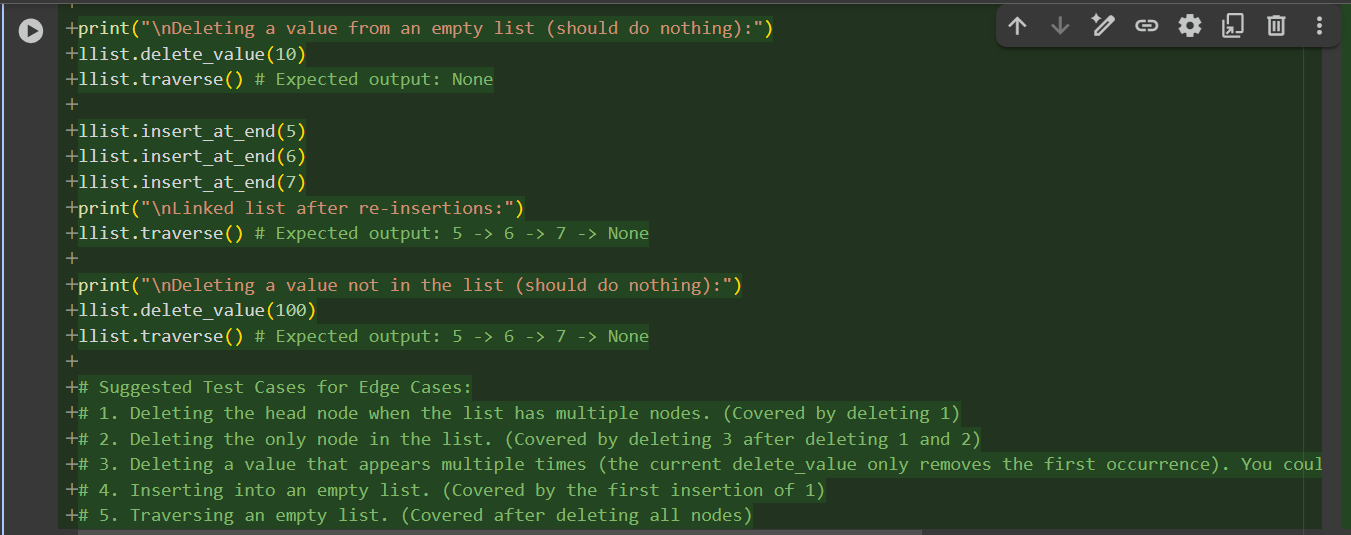
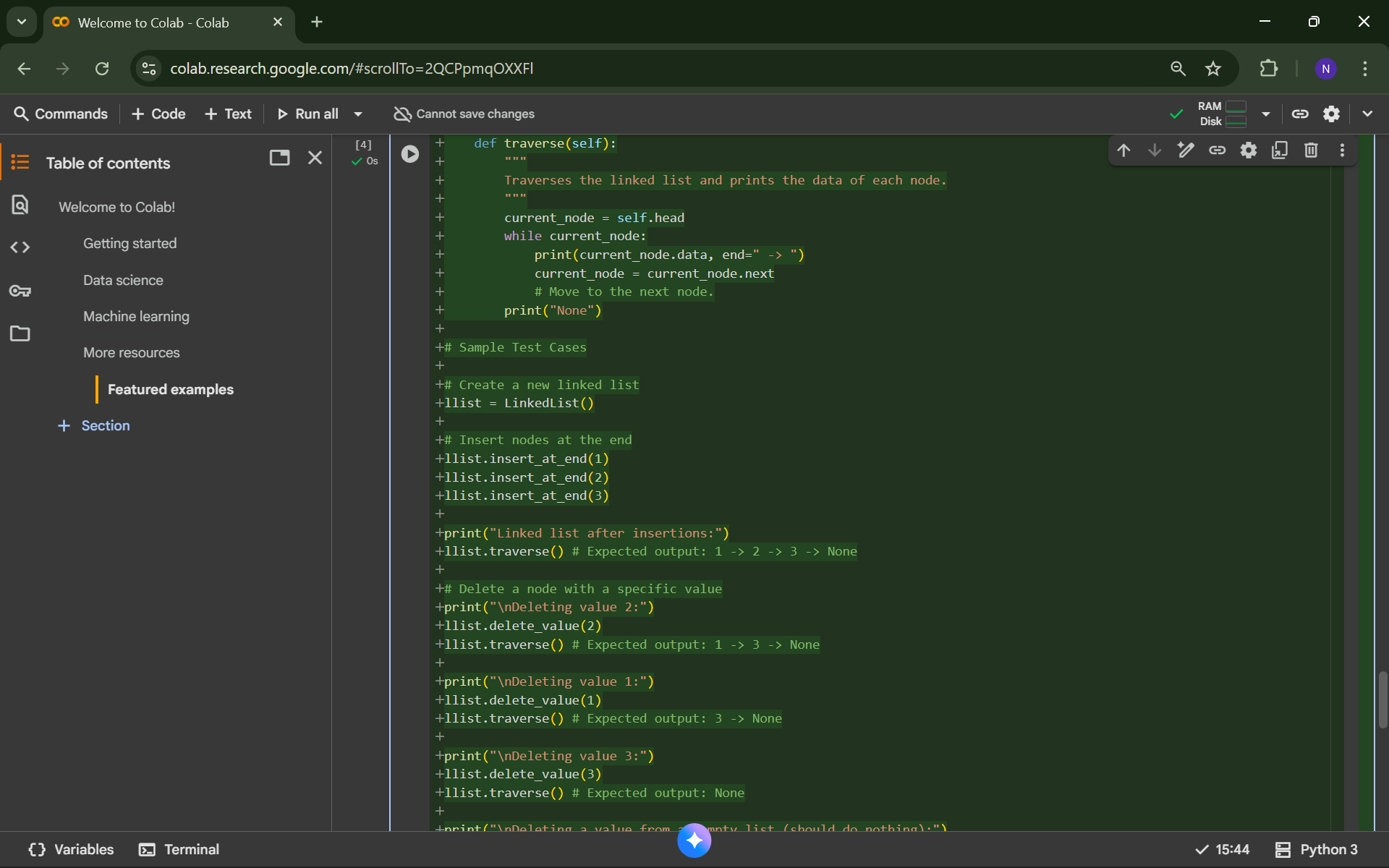
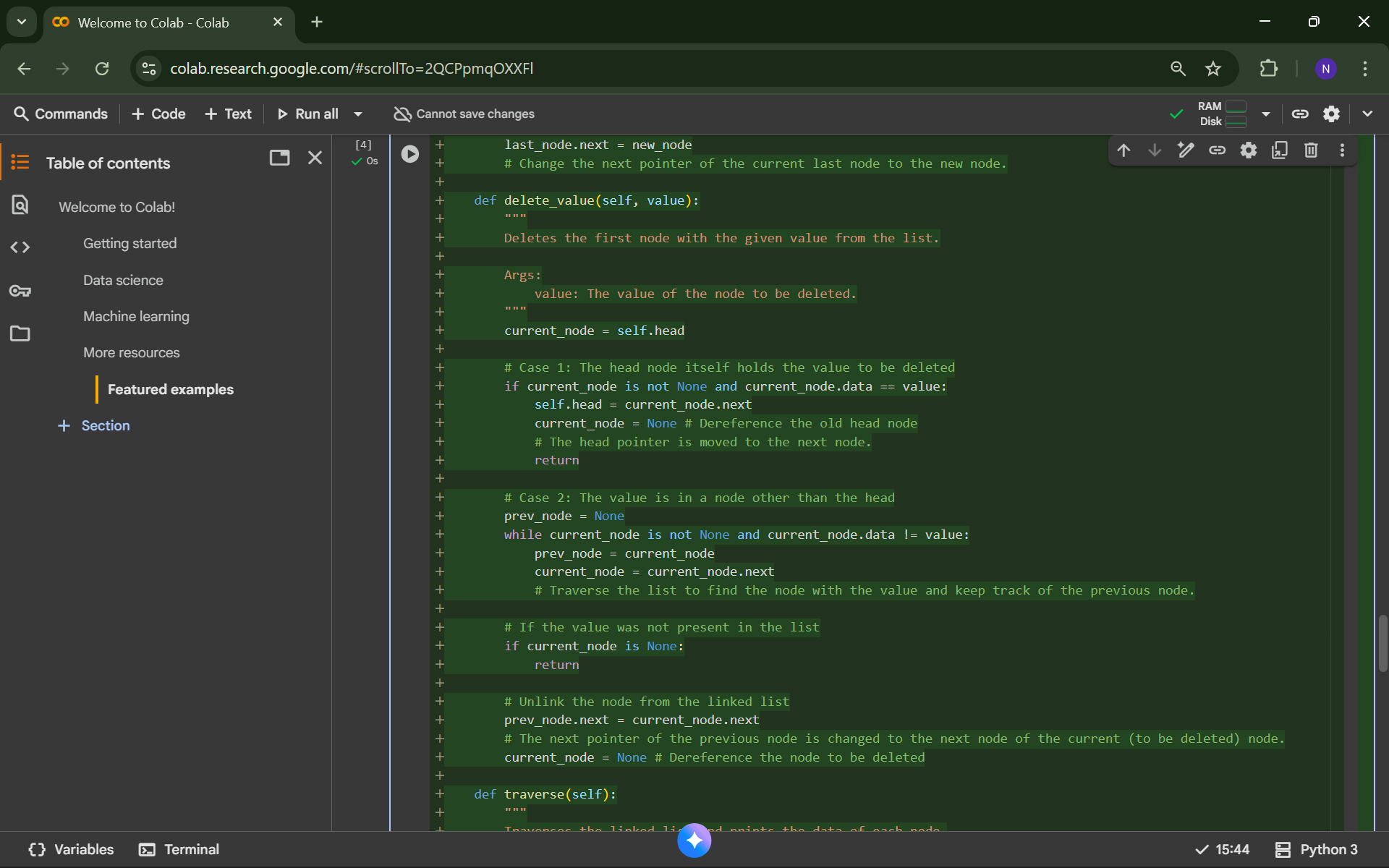
**Task 3: Singly Linked List with Traversal**

**PROMPT:**

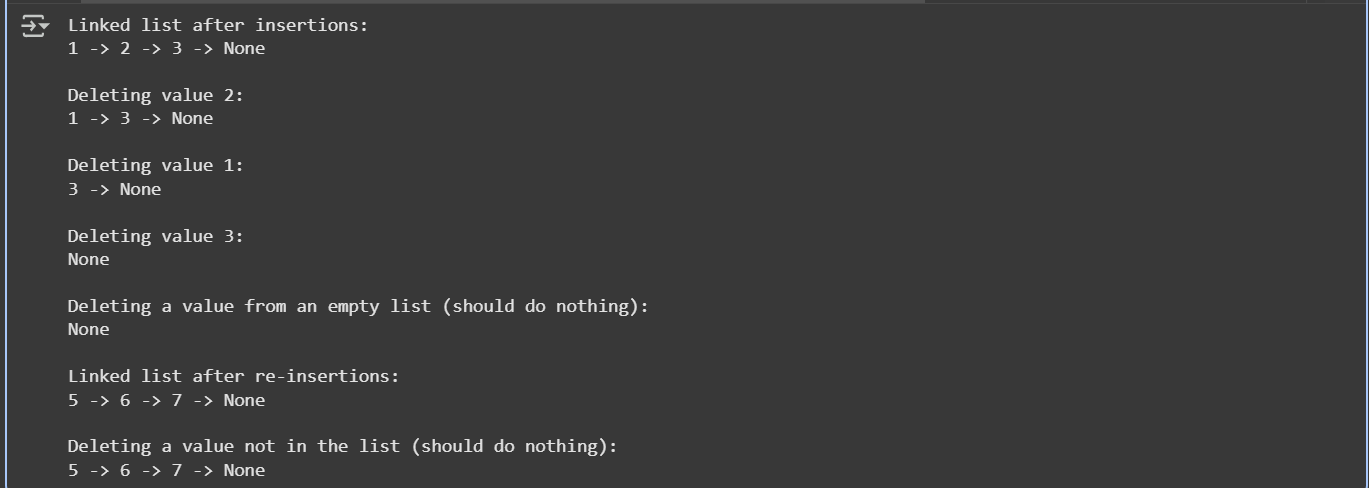
1. Create a Python Singly Linked List (Node and LinkedList) with insert\_at\_end(), delete\_value(), and traverse(). Add comments explaining pointer changes, include docstrings, and suggest test cases covering edge cases

**CODE:**





**OUTPUT:**

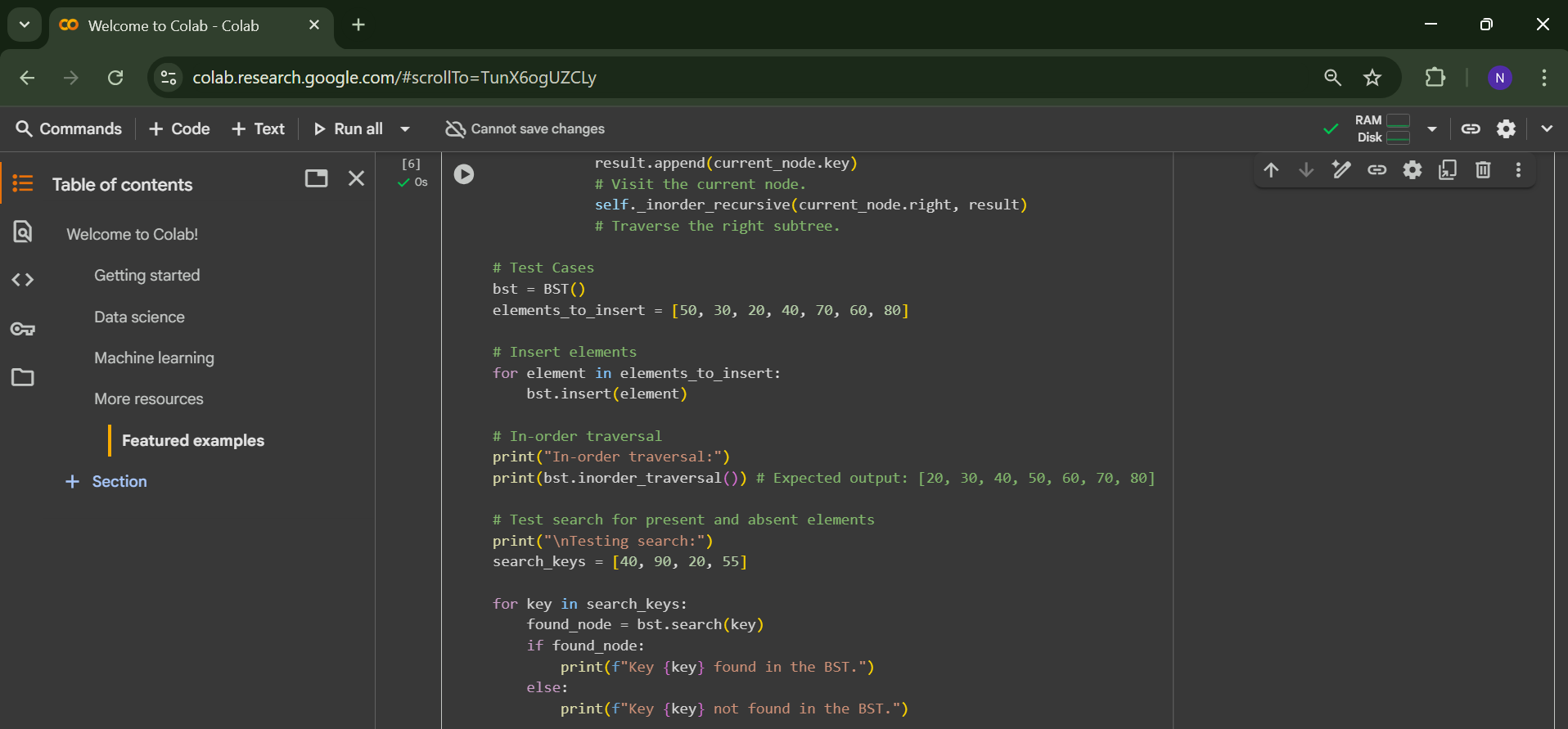
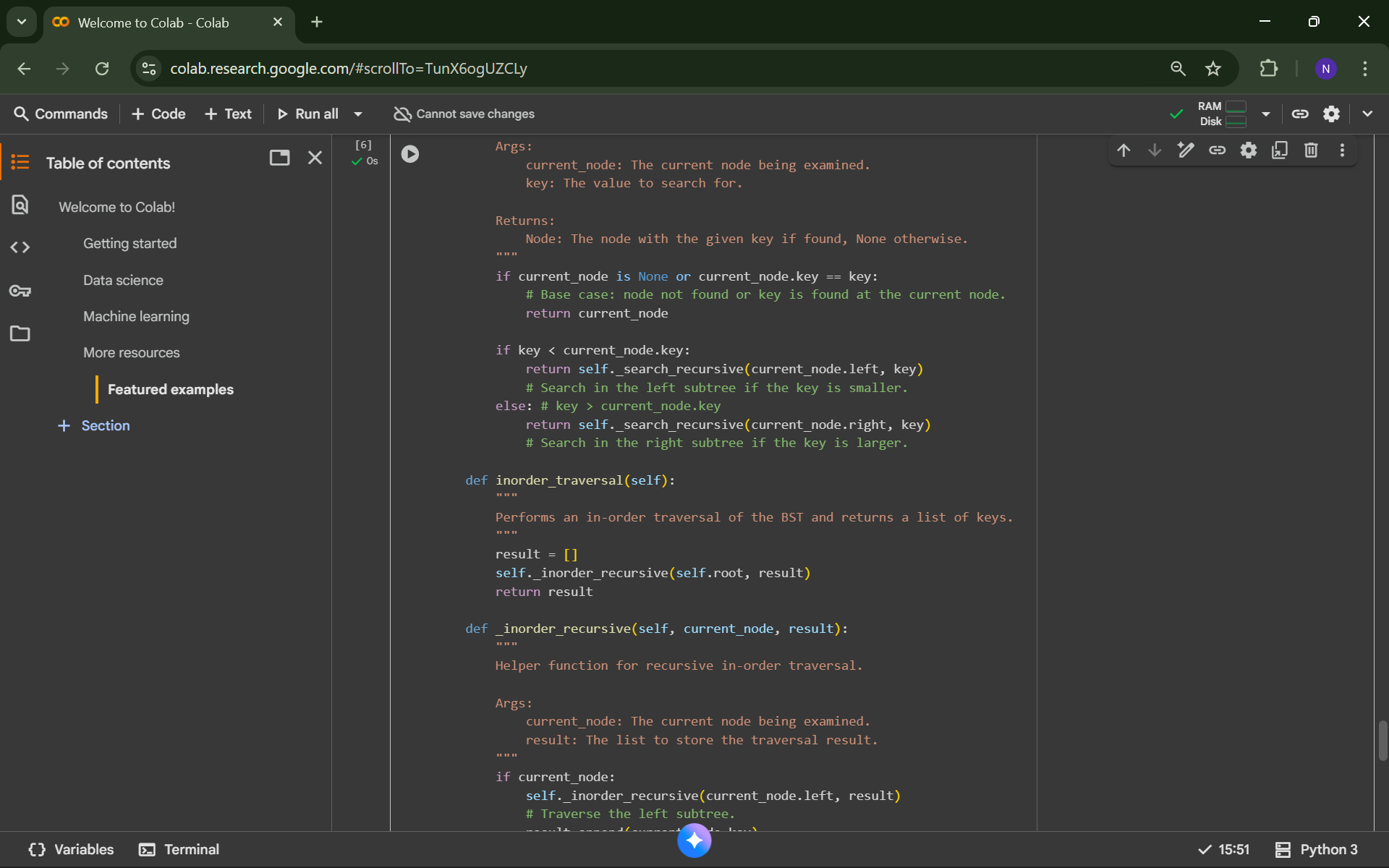
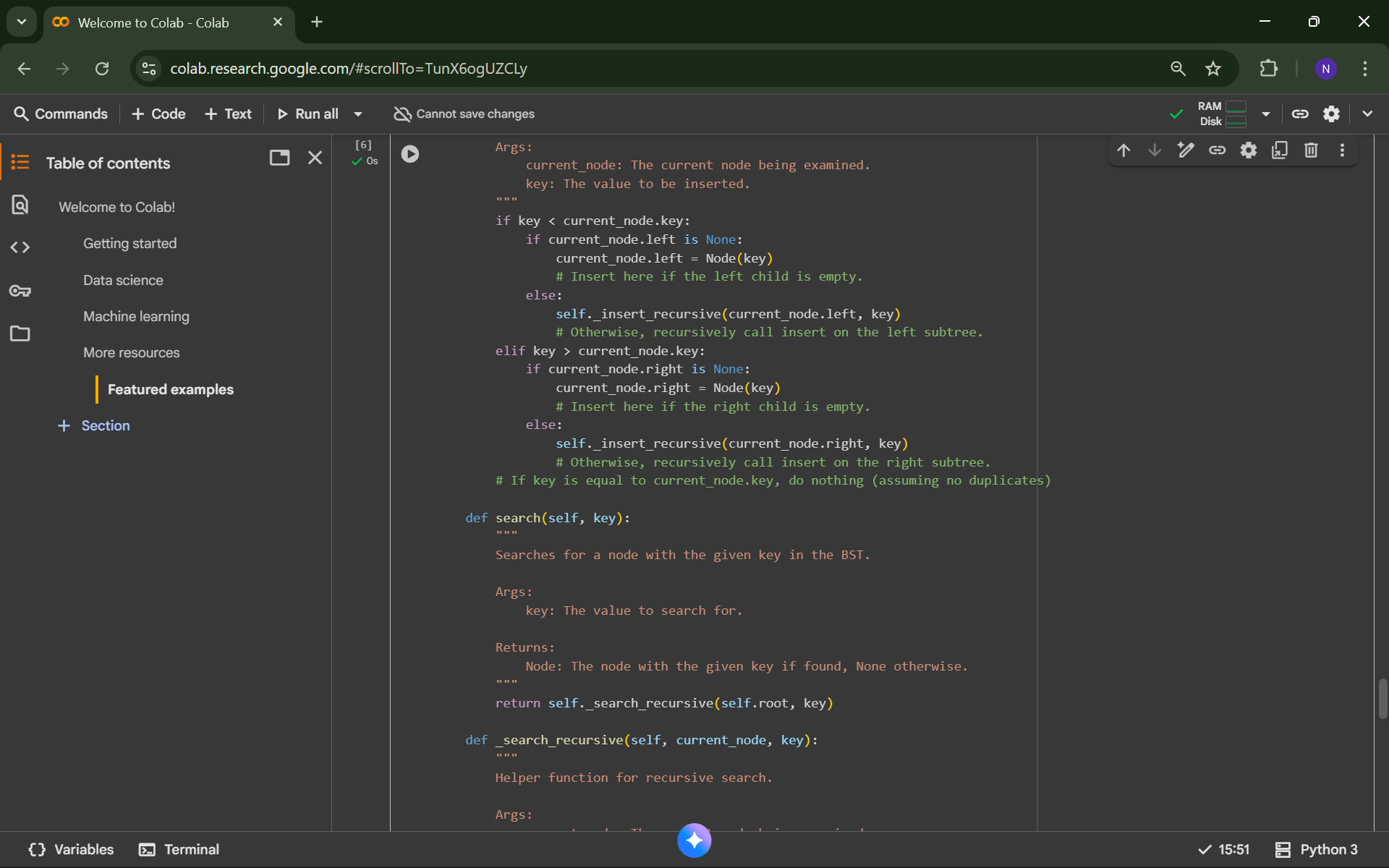
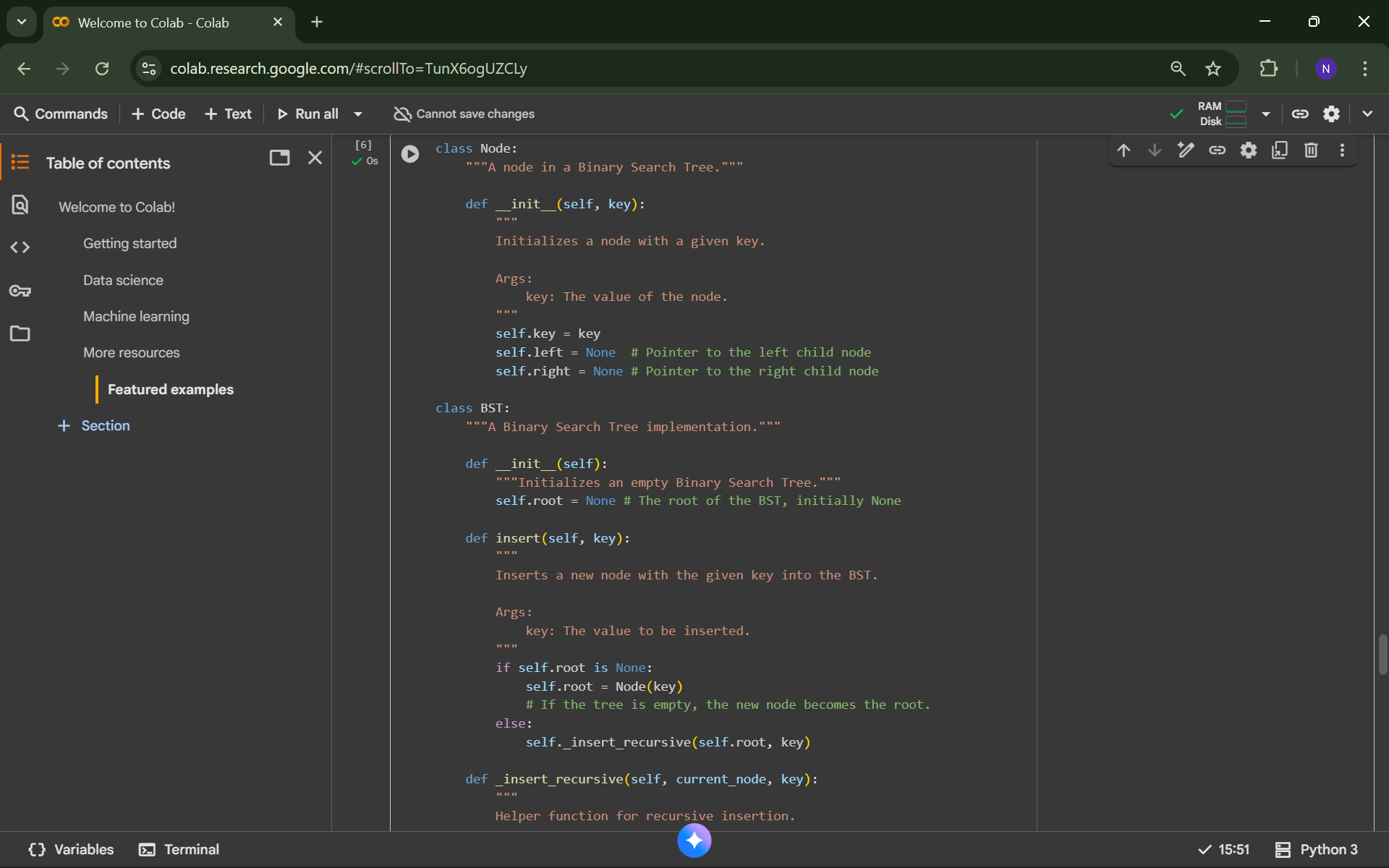


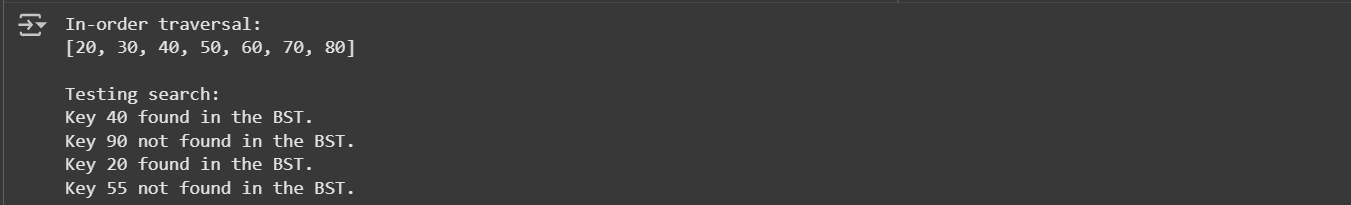
**Task 4: Binary Search Tree (BST)**

**PROMPT:**

1. Complete a Binary Search Tree with insert(), search(), and inorder\_traversal(), assuming a Node class already exists. Include docstrings, test with a list of integers, check search for present and absent values, and print an inorder traversal

**CODE:**



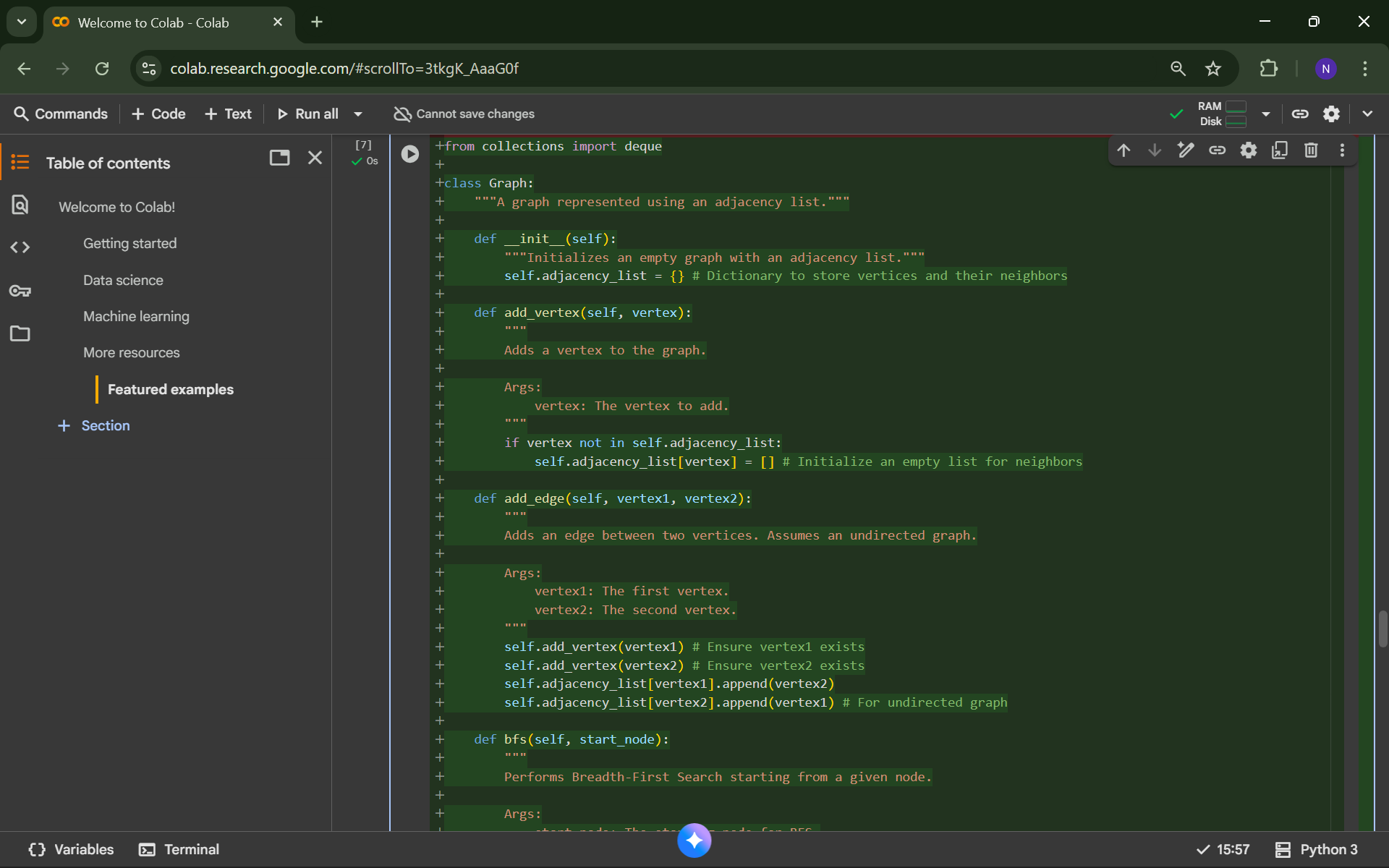
**OUTPUT:**

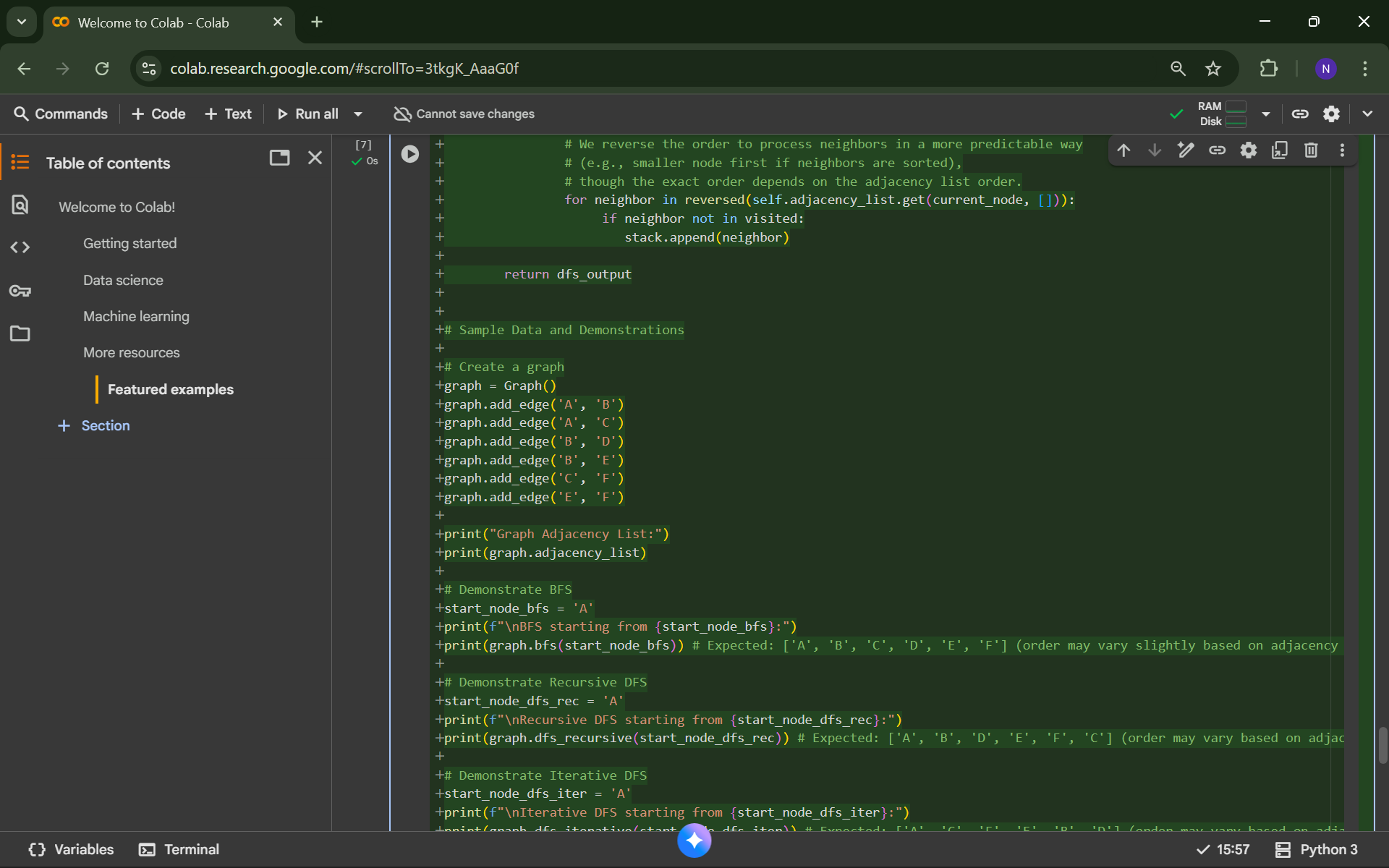
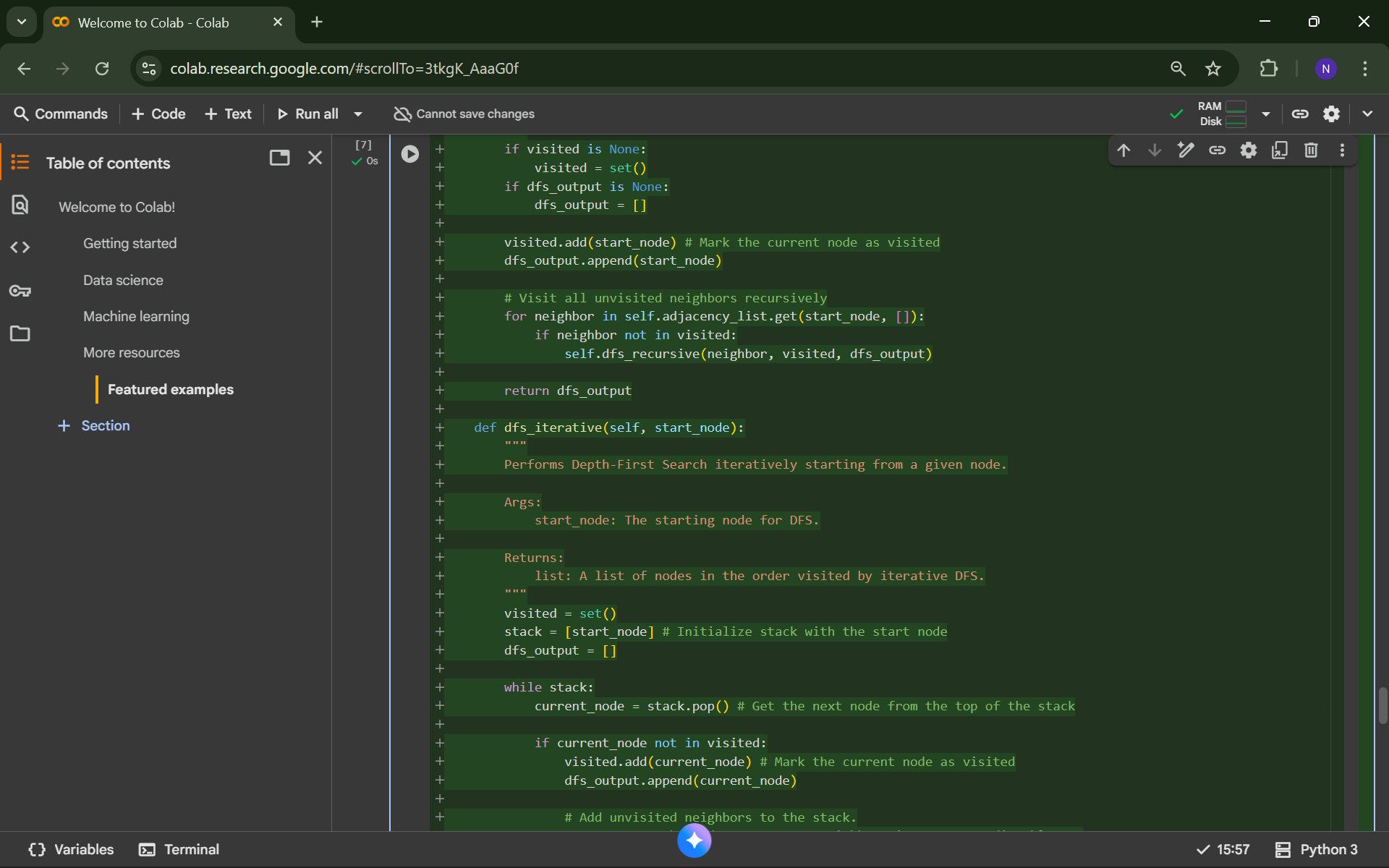
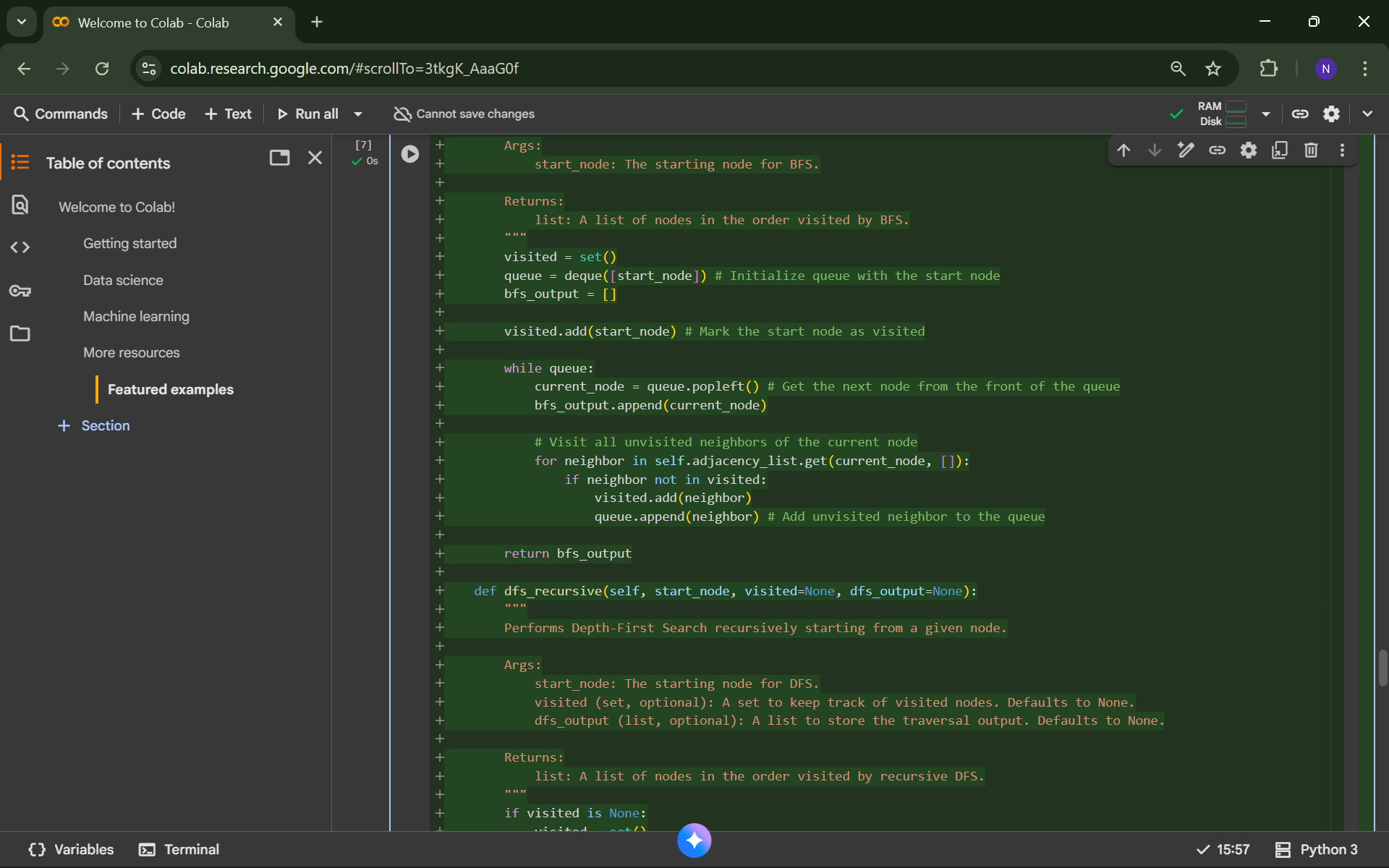
**Task 5: Graph Representation and BFS/DFS Traversal**

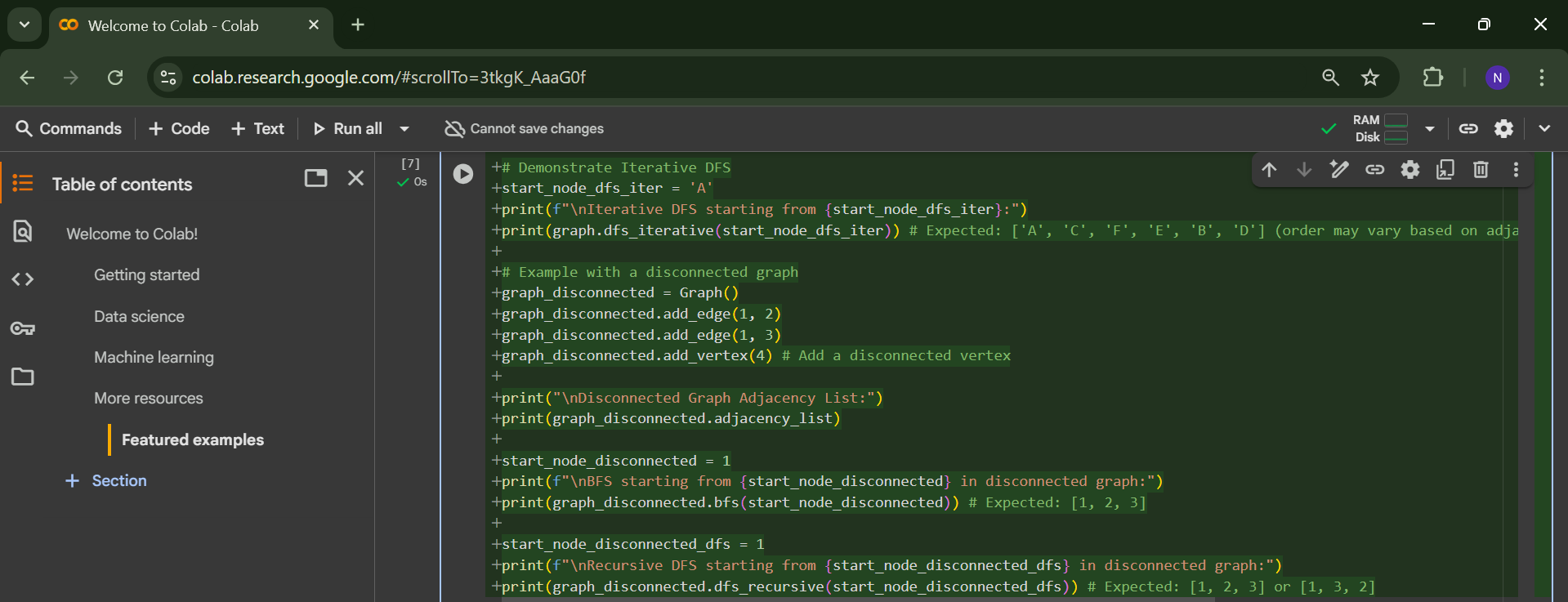
**PROMPT:**

1. Implement a Graph using an adjacency list with BFS() and DFS(). Add docstrings and comments explaining each step, show both recursive and iterative DFS, and demonstrate BFS/DFS outputs on sample data

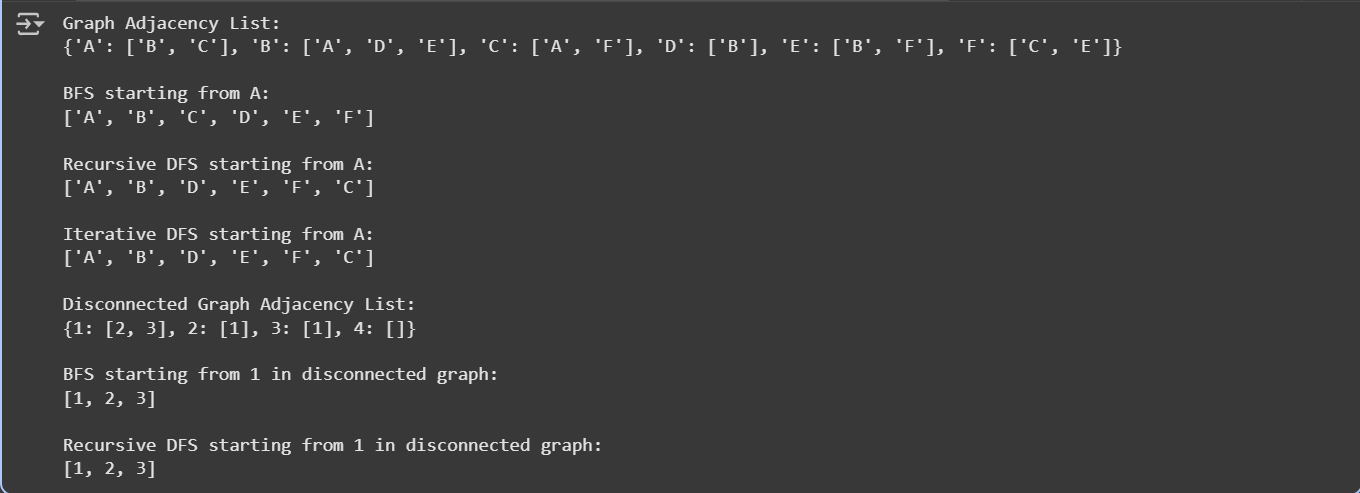
**CODE:**

****

****

****

**OUTPUT:**

****